

***THE MAPINFO USER GUIDE FOR PUSH'N'SEE***



The information in this document is subject to change without notice and does not represent a commitment on the part of the vendor or its representatives. No part of this document may be reproduced or transmitted in any form or by any means, electronic or mechanical, including photocopying without the written permission of KOREM, 680, Charest East Blvd., suite 120, Québec, Québec, G1K 3J4.

© 2007 KOREM. All rights reserved.  
Push'n'See logo is a trademark of KOREM.

Contact KOREM on the Internet at: [www.korem.com](http://www.korem.com)

KOREM Headquarters:

Tel.: (418) 647-1555  
Fax: (418) 647-1666

Toll free: 1 888 440-1MAP

**WARNING:** This software uses MapXtreme Java Edition 4.8

MapInfo Professional and MapXtreme are trademarks of MapInfo Corporation in the United-States.

Java and all Java-based trademarks and logos are trademarks or registered trademarks of Sun Microsystems, Inc. in the United-States and other countries.

This product includes software developed by the Apache Software Foundation (<http://www.apache.org>).

Products named herein may be trademarks of their respective manufacturers and are hereby recognized. Trademarked names are used editorially, to the benefit of the trademark owner, with no intent to infringe on the trademark

KOREM welcomes your comments and suggestions.



## TABLE OF CONTENT

INTRODUCTION .....	3
1. TABLES .....	5
1.1 Formats and Data Sources .....	5
1.1.1 Vectorial data.....	5
1.1.2 Raster images .....	5
1.1.3 Grids.....	5
1.1.4 DBMS Connexions (ODBC and Oracle Spatial) .....	5
1.2 Content .....	5
1.2.1 Table Names .....	5
1.2.2 Fields .....	6
1.2.3 Index .....	6
1.2.4 Seamless Tables.....	6
1.2.5 Selections and Dynamic Joins.....	7
1.3 Projections .....	7
2. WINDOWS.....	9
2.1 Map Windows .....	9
2.2 Legend Windows.....	9
2.3 Layout Windows.....	9
2.4 Graph Windows.....	9
2.5 Browser Windows.....	9
2.6 Redistrict Windows .....	10
3. MAPS .....	11
3.1 Layers Content .....	11
3.1.1 Symbols .....	11
3.1.2 Lines .....	12
3.1.3 Regions/Polygons .....	13
3.2 Layer Control Options .....	14
3.2.1 Cosmetic Layer.....	14
3.2.2 Visibility .....	14
3.2.3 Edition Mode .....	14
3.2.4 Selection Mode .....	14
3.2.5 Automatic Labels .....	15
3.2.6 Display Options .....	15
3.3 Thematics .....	16
3.3.1 Step 1 of 3: Type .....	16
3.3.2 Step 2 of 3: Data source .....	16
3.3.3 Step 3 of 3: Parameters .....	17
3.3.3.1 Objects Style .....	17
3.3.3.2 Style and Settings Options.....	18
3.3.3.3 Legend.....	21
3.4 Labels.....	21
3.4.1 Source .....	21
3.4.2 Display .....	22
3.4.3 Text Style .....	22
3.5 Projections .....	23
4. SPECIFIC FUNCTIONS .....	25
4.1 Hotlinks .....	25
4.2 Send a Workspace to Push'n'See.....	27
4.2.1 Send the Map Window to Push'n'See .....	27
4.2.2 Publication Process .....	28



## INTRODUCTION

Here are a few recommendations addressing the MapInfo user and regarding the proper organization of a workspace<sup>1</sup> to be published with Push'n'See. While Push'n'See application do not yet support every MapInfo option and function, this document will guide the MapInfo user (Push'n'See Author) to work more efficiently and avoid a certain number of problems.

These recommendations concern three aspects of a MapInfo work session: the tables (formats and content), the windows (map windows, graph windows, browser windows, layout windows and legends) and the map itself (layers, thematics and labels). We invite you to follow these recommendations in order to optimize your project's performances and ensure the integrity of your MapInfo® workspace.

We invite you to pay a special attention to the following 10 items:

1. Avoid the use of spaces, accented or special characters in file and column names;
2. Use native MapInfo tables or Vertical Mapper grids;
3. Use no selections or dynamic links;
4. Use tables of point, line or region objects;
5. Use properly registered raster images;
6. Reduce as much as possible the number of different projections used in your maps;
7. Display a map window which dimensions correspond as closely as possible to the application's, to preserve your map's interactivity;
8. Avoid storing information in the cosmetic layer. If needed, save it to a MapInfo native table;
9. Make effective use of zoom layering on layers and labels;
10. Use automatic labelling, based on simple expressions;

---

<sup>1</sup> The word "workspace" designates a MapInfo work session, i.e. the .WOR file containing the description of the work session, including each opened table or window along with their display attributes and layout settings.



## 1. TABLES

Tables opened and used by the workspace must absolutely follow these few rules:

### 1.1 Formats and Data Sources

#### 1.1.1 Vectorial data

- MapInfo tables should be composed of **.tab**, **.map**, **.id** and **.dat** files, with or without an **.ind** (index) file. Only **.dat** descriptive data file formats should be used. Other formats such as **.txt**, **.wk1**, **.xls** or **.mdb** should never be associated with the workspace.
- ESRI Shape files (**.SHP**) may be published individually in the Geobase, and then shares and used in projects. However, they cannot be included in a MapInfo workspace published in the Project module.

#### 1.1.2 Raster images

- Raster images must be properly registered and accompanied by a **.TAB** file.
- The raster images supported formats are:  
**.JPG**, **.GIF**, **.BMP**, **.PNG**, **.TIFF** (256 indexed colors), **.ECW** and **MrSID**.
- Use of **.ECW** images is not compatible with the anti-aliasing. The value of the `com.mapinfo.render.quality` property for a project using one or more ECW images should be set to « `false` ». Refer to the corresponding section of the Management Console Guide to obtain more information on this property.

#### 1.1.3 Grids

- The MapInfo thematic grids (**.mig**) are fully supported. Values associated to these theamics may be displayed when using the Push'n'See Info and Legend tools.
- The Vertical Mapper numerical grids are supported (**.GRD**), but with some restrictions.
  - When using the Push'n'See Info tool, the grids' associated values may be displayed, but specific set ups are necessary. In fact, after publishing, you must set this layer to be selectable using the Push'n'See layer Control, then save this project. You shall also ensure that no line or point objects layer is displayed directly on top of your grid.
  - The Push'n'See Legend tool does not permit to display a Vertical Mapper raster grid (**.GRD**).
- The Vertical Mapper classified grids are supported (**.GRC**), but with some restrictions. In fact, display of the values associated to this type of grid (**.GRC**) is not possible with the Push'n'See Info and Legend tools.

#### 1.1.4 DBMS Connexions (ODBC and Oracle Spatial)

- Connexions to databases (ODBC or Oracle Spatial, linked or dynamic tables) defined from your work station will not be valid when transferred on the Push'n'See server. To use a database connexion, you shall define your connexion from Push'n'See. Refer to the corresponding section of the Management Console Guide for further information.

## 1.2 Content

### 1.2.1 Table Names

As usual, file names must only contain alphanumerical characters (no accent, space or special character). They should also never start with a number or be composed of more than 31 characters.

## 1.2.2 Fields

- Field names must only contain alphanumerical characters. Do not use any accent, space or special character;
- LOGICAL fields are not fully supported. In fact, although a workspace containing one or many fields of logical type can be published without problem, such type of data should not serve as a data source for a MapInfo thematic map. They would not be displayed in Push'n'See. On the other hand, such fields can be used to create an online thematic (Push'n'See's Thematic Map Wizard).

## 1.2.3 Index

It is important to index the fields that may be used with the Push'n'See **Map Data** function. For this function, the join itself can only be performed using an indexed field.

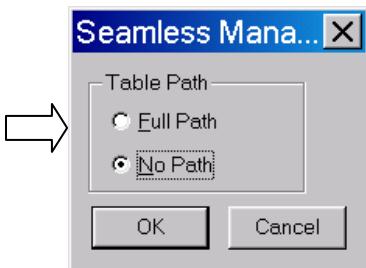
On the other hand, it is not recommended to index too many fields since the .IND file can rapidly become very heavy and thus significantly increase the publication time, as well as take up valuable space on the server. Then it is best to only index fields which are necessary for the use of the Map Data function.

## 1.2.4 Seamless Tables

Seamless tables are supported following the restrictions hereafter:

When creating seamless tables:

- All base tables composing the seamless table should be saved in the same directory as the seamless table.
- The "*No Path*" option must absolutely be checked in the Seamless Manager Preferences.



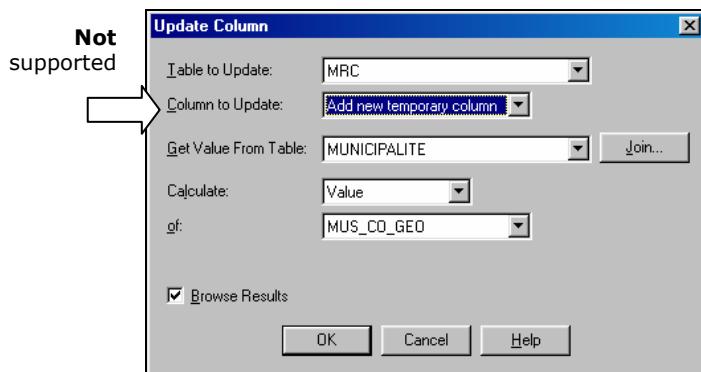
When using seamless tables:

- Push'n'See will be unable to use a seamless table if its base tables are not present in the MapInfo workspace. In consequence you must open them in a "*No View*" mode. Proceeding this way will ensure that even though they will remain invisible in the layer control, they can be called individually at any time with the Info tool.



### 1.2.5 Selections and Dynamic Joins

- SQL queries are not supported. If you wish to keep your queries results and use them in your Push'n'See projects, they should be saved in a MapInfo native format (\*.TAB). Use MapInfo's **File/Save Copy As** option.
- A table must not contain any column resulting from a dynamic join between two tables or fields. Such a column is only temporary and cannot be used by Push'n'See. When using the **Table/Update Column/Join** function, make sure to update a pre-existing column and to save this new column in making a copy of the table (**File/Save Table** or **Save Copy As**).



### 1.3 Projections

It is important to reduce the number of different projections used for the various tables layered in a map. The more different projections are used, the more time will be necessary to display the map. Moreover, the use of too many projections can, at greater scale, reduce the display precision. This can be observed both in MapInfo and Push'n'See.



## 2. WINDOWS

It is important that only one map window be opened at a time in the workspace while Push'n'See will use only one. Please note that Push'n'See considers as opened even a reduced window (  Qcmun Map  ) and will select by default the first map window it encounters in the .WOR file.

### 2.1 Map Windows

To preserve the appearance of your MapInfo map, you must make sure that your map window's width and height correspond to Push'n'See's map window's size as defined during configuration of the application (**Project Module > Design Section > Map Properties**). You should always keep in mind that the view you see in your map window when saving the workspace will be the initial view at start-up in Push'n'See. Although Push'n'See remembers the MapInfo view settings (zoom level and x,y center coordinates), these parameters are related to the window's size. In consequence of which, the effect will not be the same if the window's size is different, and your project's initial view might be entirely different. Make sure that your MapInfo map window's size is adjusted to the Push'n'See parameters in order to avoid this.

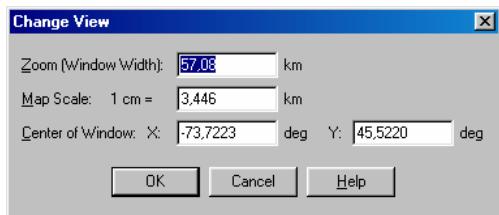
To adjust with precision your map window's width and height, you can type a line of command of this form in your MapBasic window:

Set Window FrontWindow() Width x Units "pt"

Set Window FrontWindow() Height x Units "pt"

(Point units quite resemble pixel units as used in Push'n'See)

The **Map/Change view** option and the  button in the toolbar can both allow you to precisely set the(x,y) center of your map window and the zoom level.



### 2.2 Legend Windows

Both the cartographic and thematic legends are supported by the application. You will need to specify which one is being used by your workspace (project) in the **Design / Map Properties** section of the **Project Module**.

### 2.3 Layout Windows

Layout windows are not supported by the application.

### 2.4 Graph Windows

Graph windows are not supported by the application.

### 2.5 Browser Windows

Browser windows are not supported by the application.

## **2.6 Redistrict Windows**

Redistrict windows are not supported by the application.

## 3. MAPS

### 3.1 Layers Content

- Layers can contain any type of points (symbols) , lines  or regions .
- The tables should contain no text object . This, of course, does not apply to automatic labelling.
- Objects collection:

	Supported	Not supported
Combination of polygons/regions	<b>X</b>	
Combination of lines/polylines	<b>X</b>	
Combination of points		<b>X</b>
Combination of different types of objects		<b>X</b>

#### 3.1.1 Symbols



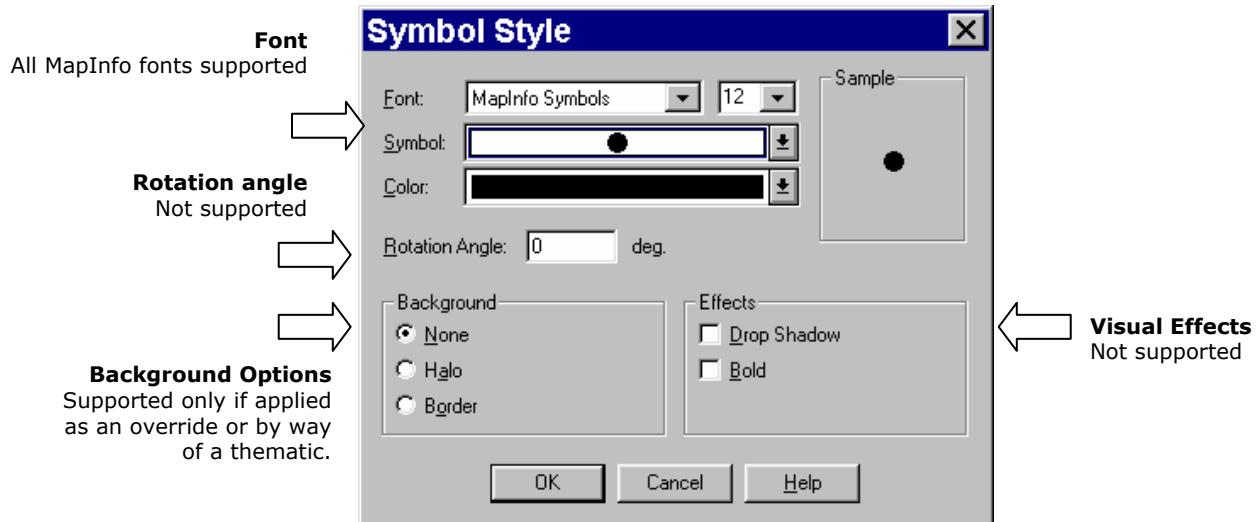
##### Symbol Types

- All *MapInfo* symbols are supported;
- Custom symbols can be used by the application only if they are applied in the workspace as Style Override (Menu Map > Layer Control > Display > Style Override) or by way of a thematic (Menu Map > Create Thematic Map). Make sure they have been uploaded on the server in a **.gif or .jpg** format. Ask the System Administrator;



##### Symbol Style

- Backgrounds (borders and shadings) are supported only if applied as an override (Map menu > Layer Control > Display > Style Override) or by way of a thematic (Map menu > Create Thematic Map);
- Some attributes and visual effects are not supported:
  - ✓ Rotation angle;
  - ✓ Drop shadow;
  - ✓ Bold.

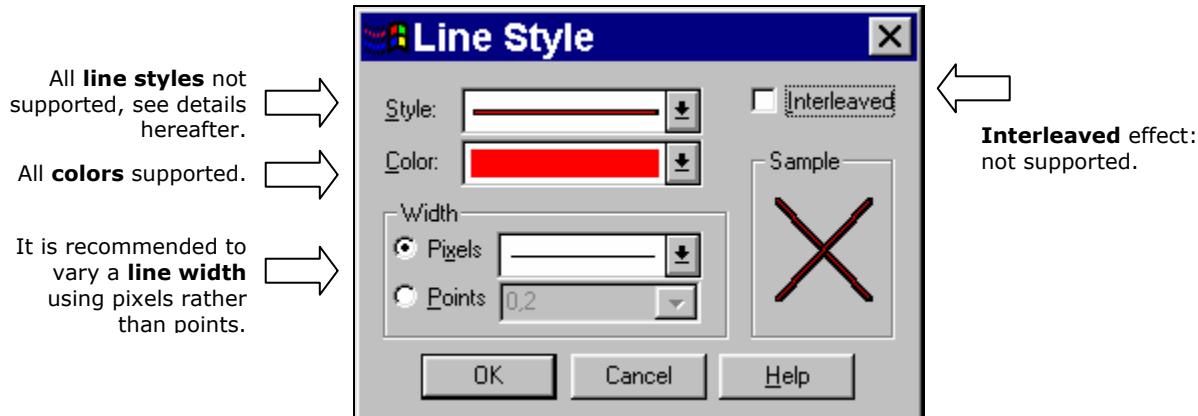


### 3.1.2 Lines

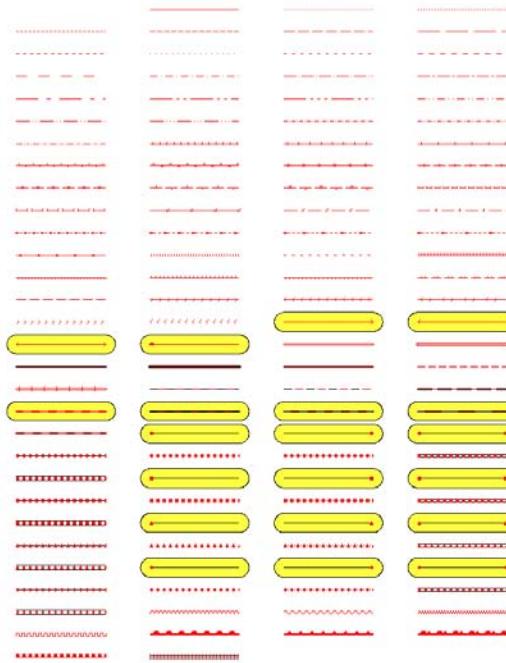


#### Line Style

Linear objects can be of any color and width. All styles are not supported. See details hereafter.



For a line width of 1 pixel, the designated line styles are not supported. Warning: The more complex a line gets, the more chances there are that it will lose display quality as the width of the line increases.

**Legend:**

 : Unsupported line style

### 3.1.3 Regions/Polygons



#### Regions Geometry

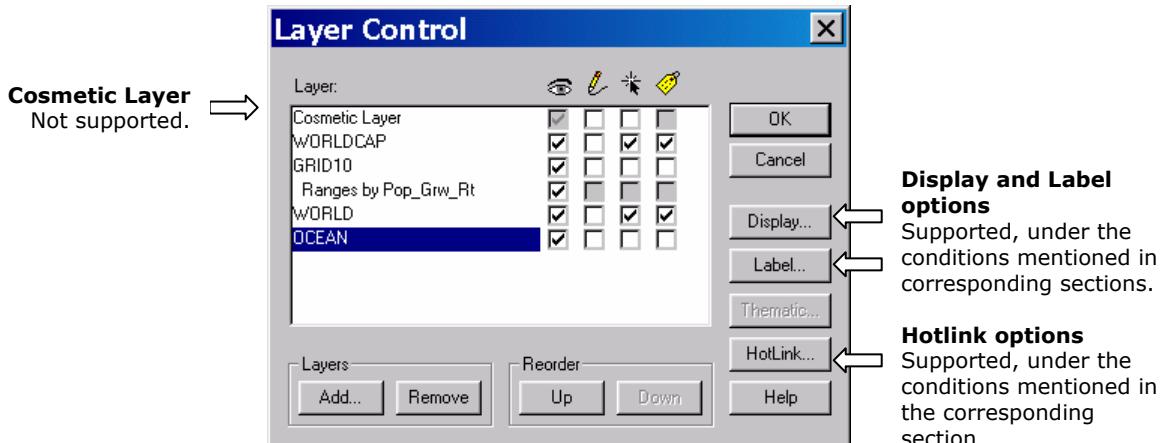
The more complex a region (polygon) gets, the more it can cause undetermined effects to the display quality and lengthen the display delay. The use of MapInfo 6.5 Generalize function or of the MapLogix topological map editing utility can be a solution to simplifying polygonal objects.



#### Region Style

- All fill patterns and colors are supported.
- Border styles follow the same restrictions as line styles (See previous page).
- Transparency effect is fully supported all the time.
- Use of fill patterns with or without transparency is supported all the time.
- Border styles follow the same restrictions as lines (see corresponding section of this guide).

## 3.2 Layer Control Options



### 3.2.1 Cosmetic Layer

The cosmetic layer is not supported by the application. It must be kept cleared and not be in Edition mode.

### 3.2.2 Visibility



Layers visibility parameters will remain as defined in the MapInfo workspace, including the zoom layering settings.

Note that proper use of zoom layering will increase the performances of a cartographic project (Display time).

### 3.2.3 Edition Mode



The Edition mode is not available within the application. It is recommended not to keep any layer in the Edition mode.

### 3.2.4 Selection Mode



This option is supported and the parameters will remain as defined in the MapInfo workspace. The following Push'n'See tools require layers to be selectable:

- ✓ Information;
- ✓ Drilldown;
- ✓ Hotlink;
- ✓ Selections.

Even though the selection option for the Vertical Mapper numerical grids (format .GRD) is not available in the MapInfo layer control, this option may be activated in Push'n'See's layer control to allow the use of the Info tool on this type of files.

### 3.2.5 Automatic Labels

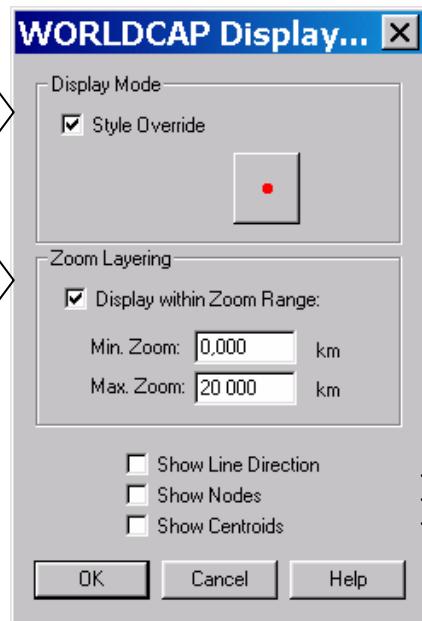


Automatic Labels are supported and will respect the labels parameters under the further mentioned conditions. See the corresponding section of this guide.

### 3.2.6 Display Options

**Style Override options**  
Supported, under the further mentioned restrictions, according to each object type. See the corresponding section of this guide.

**Zoom layering**  
Supported.



**Show line direction, nodes and centroids**

Options not supported.

## 3.3 Thematics

### 3.3.1 Step 1 of 3: Type

Push'n'See supports the following types of thematic representations:



Ranges;



Bar Charts;



Pie Charts;



Graduated Symbols;



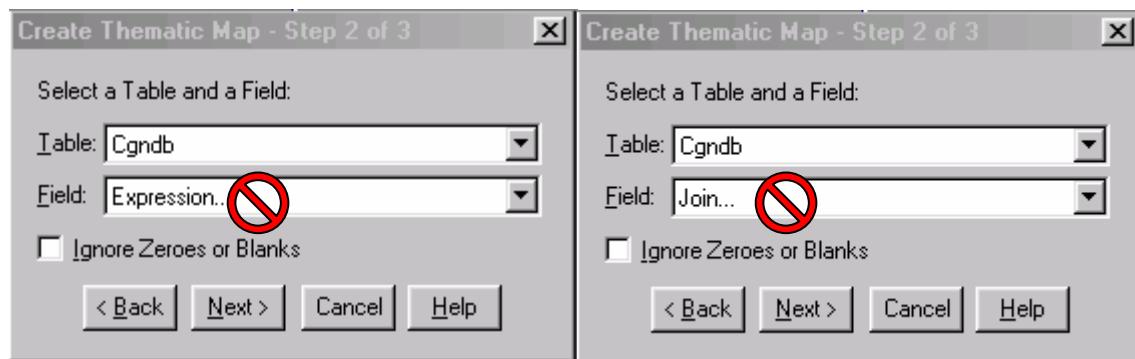
Individual Values;



Grid.

### 3.3.2 Step 2 of 3: Data source

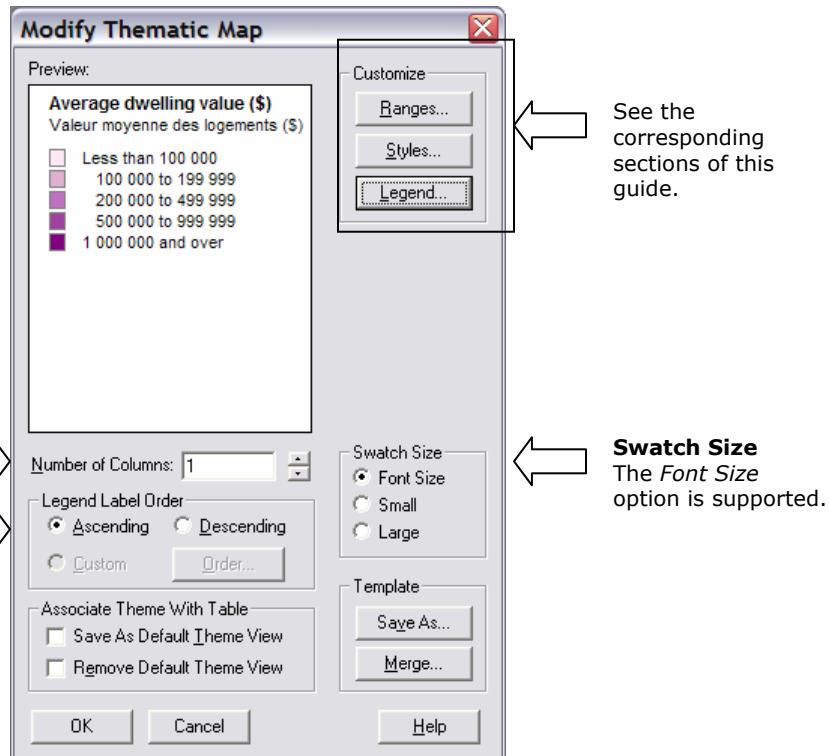
- Logical fields must not be used in the creation of any kind of MapInfo thematic maps. It would not be displayed in Push'n'See. On the other hand, one will be able to create an on-line thematic using the Push'n'See Thematic Map Wizard.
- A thematic should not refer to anything else but a unique field. Operations between tables or fields (*Expression...* or *Join...*) are not recognized by the application.



- References to an ODBC (Open Data Base Connexion) should not be used.
- Bivariated theatics are not supported by the application.

### 3.3.3 Step 3 of 3: Parameters

- Number of columns: Legend display in only one column is supported. Legend display in two or more columns is not supported.
- Display order of legend elements: only the ascending and descending modes are supported. Definition of a custom order is not supported.
- Swatch Size : Only the default value – Font Size – is supported.



#### 3.3.3.1 Objects Style



##### Symbols:

The same restrictions apply as described in the previous section.



##### Lines:

The same restrictions apply as described in the previous section.



##### Regions:

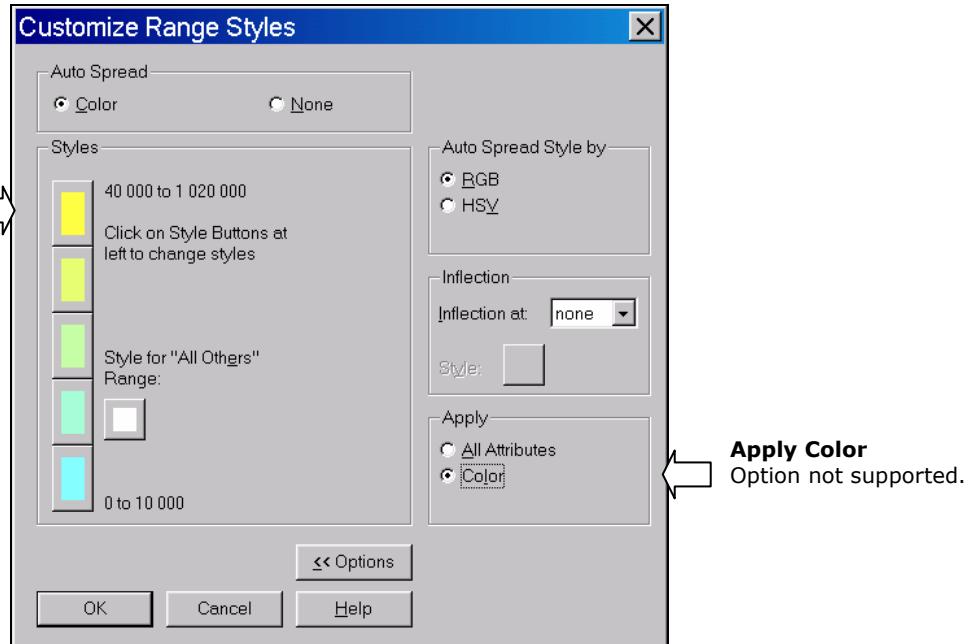
The same restrictions apply as described in the previous section.

### 3.3.3.2 Style and Settings Options



#### Ranges

The following restrictions apply:





### Bar Charts

The following restrictions apply:

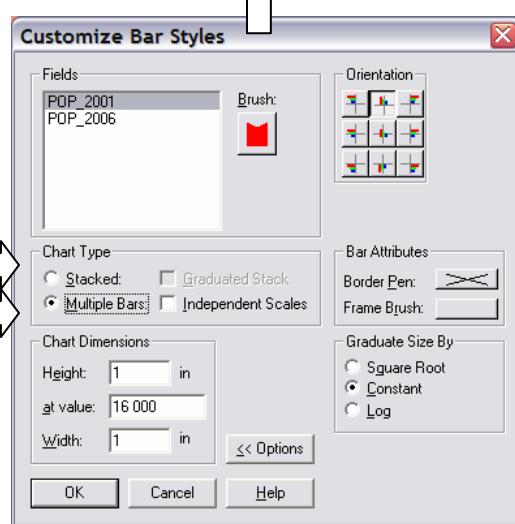
**Style:** Supported, under conditions mentioned in previous sections.

**Stacked data :**  
Option not supported.

**Independent scales :**  
Option not supported.

**Pen and brush style:** Options not supported.

**Logarithmic Transformation:**  
Option not supported.



### Pie Charts

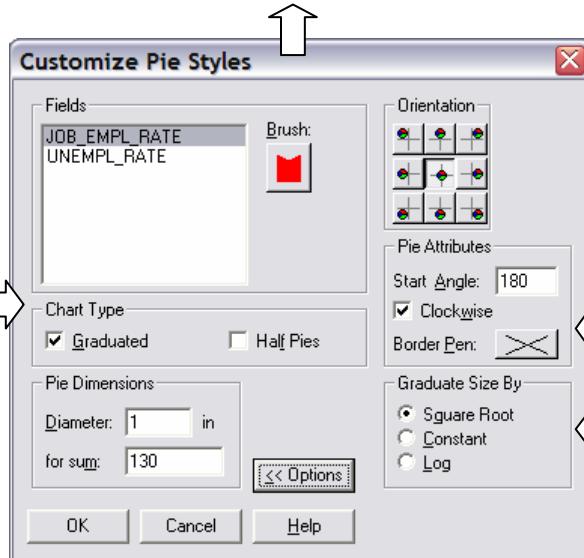
The following restrictions apply:

**Style:** Supported, under conditions mentioned in previous sections.

**Half Pies**  
Option not supported.

**Border style**  
Only the default value is supported (black border, 1 pix.).

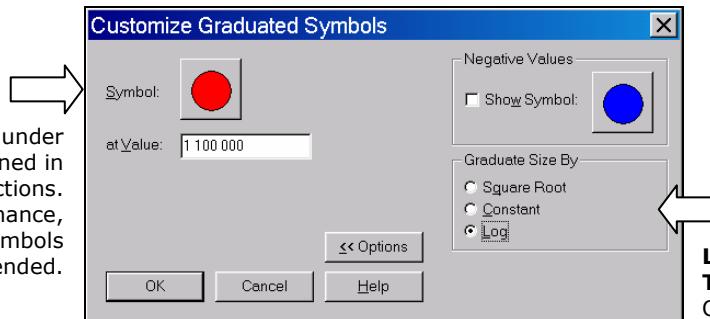
**Logarithmic Transformation**  
Option not supported.





### Graduated Symbols

The following restrictions apply:



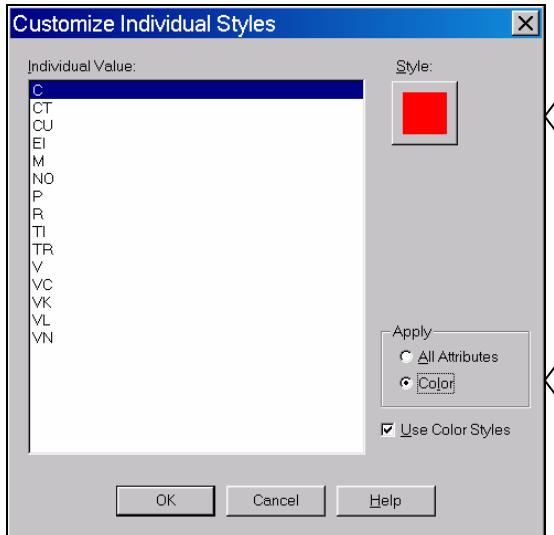
**Style:** Supported, under conditions mentioned in the previous sections. For better performance, the use of MapInfo symbols is recommended.

**Logarithmic Transformation**  
Option not supported.



### Individual Values

The following restrictions apply:



**Style**  
Supported, under conditions mentioned in previous sections.

**Apply Color**  
Option not supported

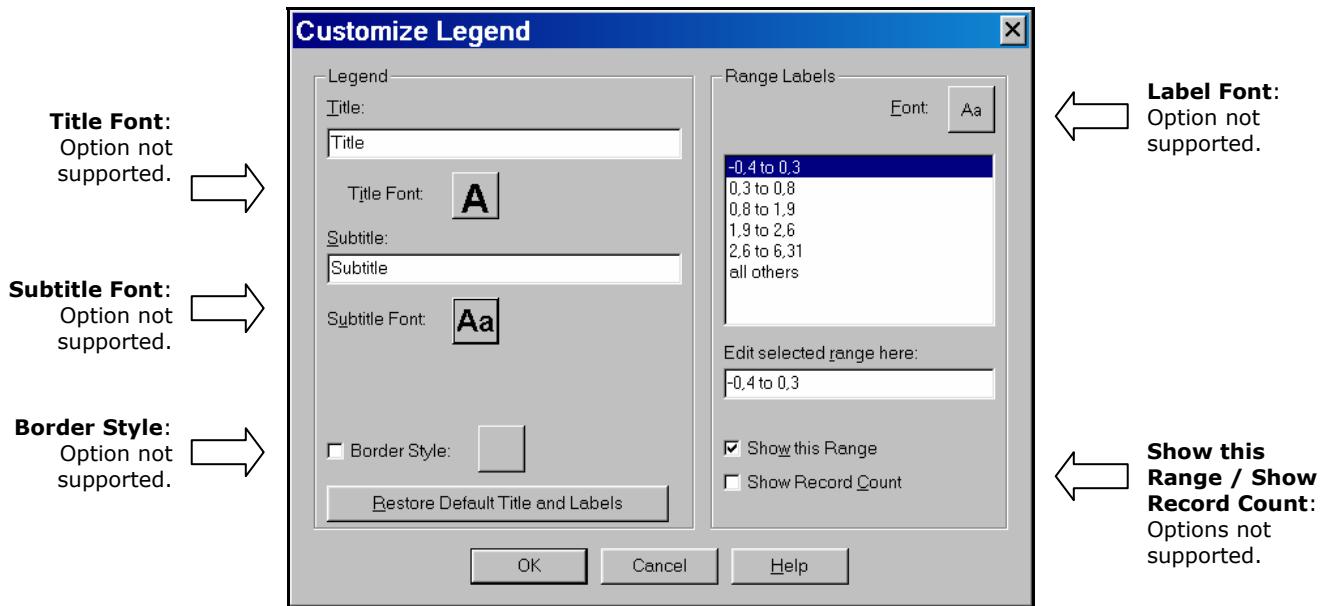


### Grid

The following restrictions apply:

- **Interpolation mode:** Both IDW and TIN modes are supported. On the other hand, legend values of an IDW interpolation will not be displayed properly.
- It is important to make sure not to modify the original thematic since any subsequent modification with the Modify Thematic Map option of the Map menu will be ignored. In order to modify a grid and be sure that all the data will be kept during the transfer to Push'n'See, the .GRID file must be created anew by starting the interpolation process over entirely (Map Menu > Thematics).

### 3.3.3.3 Legend



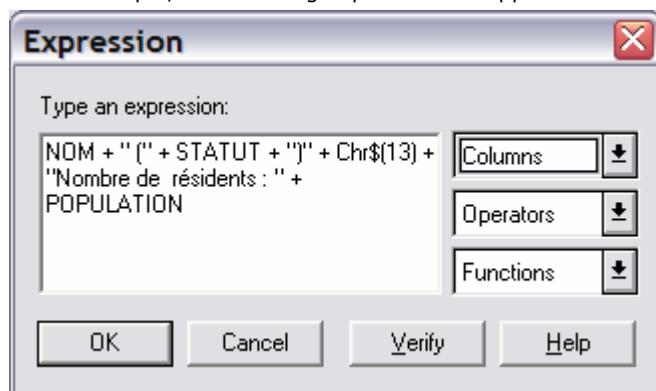
## 3.4 Labels

### 3.4.1 Source

The label source may be :

- A field;
- A simple expression using the operators and following functions :
  - concatenation (+) between two CHARACTER fields;
  - insertion of a carriage return : Chr\$(13) ;
  - insertion of a chain of characters.

As an example, the following expression is supported:



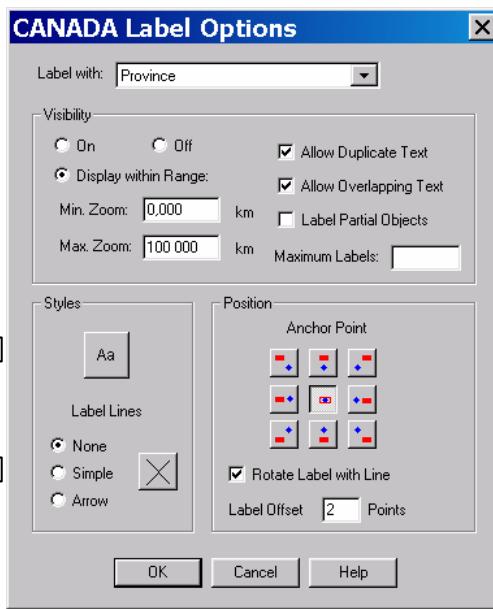
Where NOM, STATUT and POPULATION are field names.

### 3.4.2 Display

\*Note : Options not supported, unless stated differently.

These label display options are not supported :

- ✓ Max Labels;
- ✓ Label Partial Segments;
- ✓ Label Line



**Text Style**  
Restrictions apply as mentioned in the previous section.

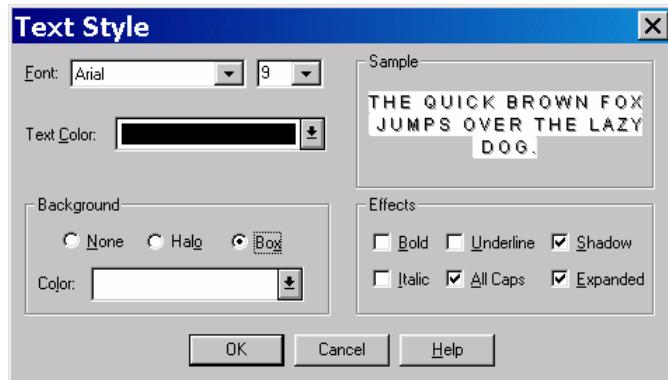
**Label Line**  
Not supported.

**Label Partial Objects and Maximum Labels**  
Not supported.

### 3.4.3 Text Style

These text styles are not supported :

- ✓ All Caps;
- ✓ Expanded;
- ✓ Shadow.



**Shadow, all caps and expanded characters**  
Not supported.

### 3.5 Projections

The projection used to display your map (*Map > Options > Projections*) should as much as possible correspond to the projection of the tables. As mentioned in a previous section, the more different projections are used, the more time will be necessary to display and redraw the map. Moreover, the use of too many projections can, at greater scale, reduce the display precision. This can be observed both in MapInfo and Push'n'See.

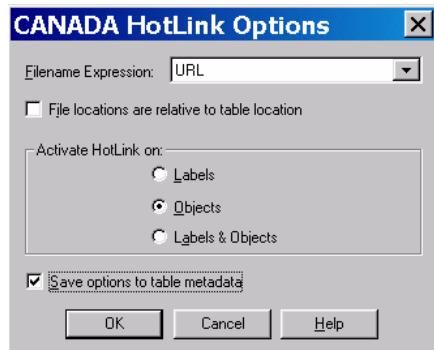


## 4. SPECIFIC FUNCTIONS

### 4.1 Hotlinks

For the application to be able to read MapInfo's hotspot function's parameters the following conditions must be observed:

1. Activate the function on objects only.
2. Check the *Save options to table metadata* option. This way, MapInfo will write the Hotlink parameters into the table's .TAB file hence allowing the application to read the necessary information.



However MapInfo will not keep more than one hotspot definition. To work around this limitation, proceed in the following manner:

1. Activate the Hotlink function on a first field, as indicated here above.
2. Open your table's .TAB file in any text editor, such as *WordPad*. The hotspot is defined in four lines under the metadata section as underlined in the following example.

```
!table
!version 300
!charset WindowsLatin1

Definition Table
  Type NATIVE Charset "WindowsLatin1"
  Fields 3
    Bureau Char (20) Index 1 ;
    Adresse Char (120) ;
    WEB Char (120) ;
  ReadOnly
  begin_metadata
    "\ActiveObject" = ""
    "\ActiveObject\Expr" = "URL"
    "\ActiveObject\Mode" = "HOTLINK MODE OBJ"
    "\ActiveObject\Relative" = "FALSE"
    "\IsReadOnly" = "FALSE"
  end_metadata
```

3. Cut and paste these four lines as many times as you have hotspots to be defined.

```
!table
!version 300
!charset WindowsLatin1

Definition Table
  Type NATIVE Charset "WindowsLatin1"
  Fields 3
```

```

Bureau Char (20) Index 1 ;
Adresse Char (120) ;
WEB Char (120) ;
ReadOnly
begin_metadata
"\ActiveObject" = ""
"\ActiveObject\Expr" = "URL"
"\ActiveObject\Mode" = "HOTLINK MODE OBJ"
"\ActiveObject\Relative" = "FALSE"
"\ActiveObject" = ""
"\ActiveObject\Expr" = "URL"
"\ActiveObject\Mode" = "HOTLINK MODE OBJ"
"\ActiveObject\Relative" = "FALSE"
"\ActiveObject" = ""
"\ActiveObject\Expr" = "URL"
"\ActiveObject\Mode" = "HOTLINK MODE OBJ"
"\ActiveObject\Relative" = "FALSE"
"\IsReadOnly" = "FALSE"
end_metadata

```

4. For each Hotlink, modify the following two elements:

- ✓ The Hotlink **identifier**, define by the `\ActiveObject\` string:

For the application to differentiate each hotlink, the identifiers must be unique. In the following example each hotlink definition is identified by a number: `ActiveObject1`, `ActiveObject2`, `ActiveObject3`.

For each Hotlink definition, make sure to modify all corresponding four lines.

Example:

```

"\ActiveObject1" = ""
"\ActiveObject1\Expr" = "URL"
"\ActiveObject1\Mode" = "HOTLINK MODE OBJ"
"\ActiveObject1\Relative" = "FALSE"
"\ActiveObject2" = ""
"\ActiveObject2\Expr" = "URL"
"\ActiveObject2\Mode" = "HOTLINK MODE OBJ"
"\ActiveObject2\Relative" = "FALSE"
"\ActiveObject3" = ""
"\ActiveObject3\Expr" = "URL"
"\ActiveObject3\Mode" = "HOTLINK MODE OBJ"
"\ActiveObject3\Relative" = "FALSE"

```

- ✓ The hotlink **field's name**, i.e. the name of the field containing the Hotlink reference (URL) and defined as follows: `"\ActiveObject \Expr"="URL_1"`, where "URL\_1" is the name of the field. In the following example, URL\_1, URL\_2 and URL\_3 each identifies a hotlink field.

Example:

```

"\ActiveObject1" = ""
"\ActiveObject1\Expr" = "URL_1"
"\ActiveObject1\Mode" = "HOTLINK MODE OBJ"
"\ActiveObject1\Relative" = "FALSE"
"\ActiveObject2" = ""
"\ActiveObject2\Expr" = "URL_2"
"\ActiveObject2\Mode" = "HOTLINK MODE OBJ"
"\ActiveObject2\Relative" = "FALSE"
"\ActiveObject3" = ""
"\ActiveObject3\Expr" = "URL_3"
"\ActiveObject3\Mode" = "HOTLINK MODE OBJ"
"\ActiveObject3\Relative" = "FALSE"

```

5. Save the table and proceed as for a regular publication.

## 4.2 Send a Workspace to Push'n'See

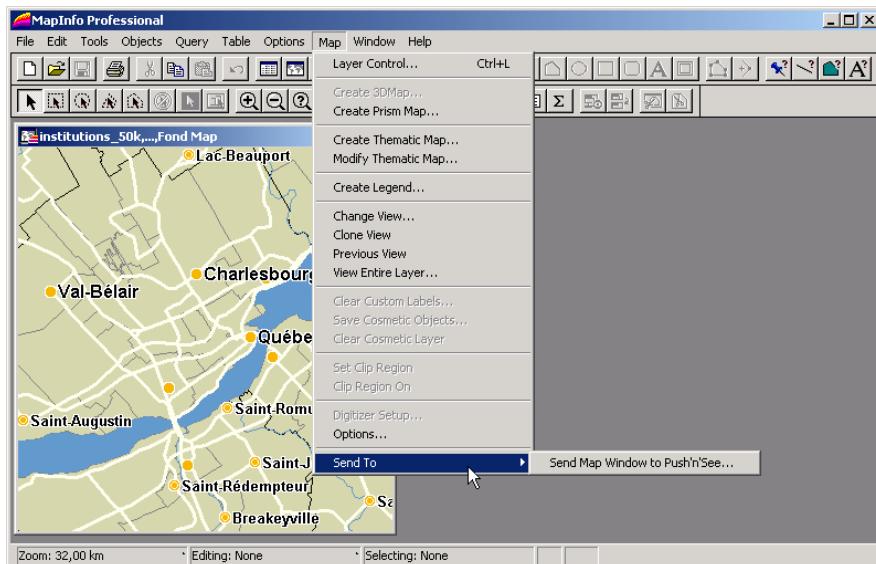
The « Send Map Window to Push'n'See » function allows the MapInfo user to send (Publish) his workspace directly from MapInfo to Push'n'See. It also integrates a workspace analysis tool which will help the user to confirm that the content and properties of his workspace are compatible with the application, and thus ensure a successful publication.

Prior to using this function, make sure of two things :

1. That the application Scan\_PNS\_en.MBX is installed in the same directory as the "Send to Push'n'See" (pns.mbx) application.
2. That only one map window is opened.

### 4.2.1 Send the Map Window to Push'n'See

The image below shows how to access the application from within the MapInfo environment: Map > Send To > Send MapWindow to Push'n'See.



When launching the application, the following window appears:

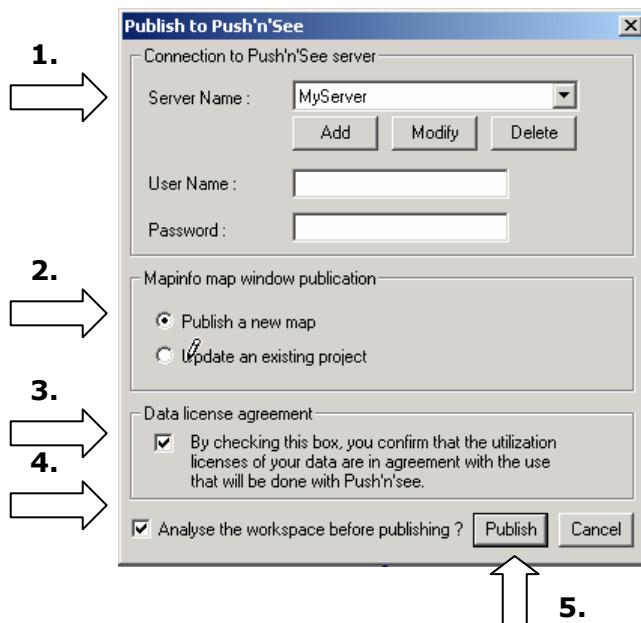


A user with an access to a Push'n'See server will be able to immediately use the full version of the application. By checking the "Don't display anymore" option, he will prevent from displaying this page in further uses of the application. He will then be redirected to the main publication page, as described below. The "Continue" button will lead him to the first step of the publication process.

A user without an access code to Push'n'See will have to use the evaluation version. By clicking on the "Evaluate during 30 days" button he will be redirected to a registration form. When completed, the user will receive a temporary user name and password from KOREM's support team. These will allow him to proceed with the publication of his MapInfo workspace, this time using the full version of the application. The process is then exactly the same as for the regular user. The temporary user name and password are valid for a duration of thirty (30) days.

#### 4.2.2 Publication Process

To launch the publication process, click on the "Continue" button in the here above window. Five easy steps are to be taken:



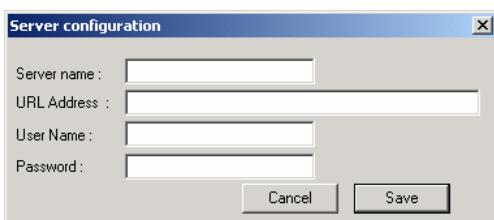
##### 1. Push'n'See server configuration

To connect to a Push'n'See server, the user must first define the following three parameters:

- ✓ The Push'n'See server itself;
- ✓ The user name to access Push'n'See;
- ✓ The password to access Push'n'See.

The « Add » button will allow to create a new server configuration.

To configure a server, the user must know the complete address to the Push'n'See publication module (URL) and own a valid user name and password. To each configured server, a name can be attributed to facilitate identification.



The URL address must have the following form:

<http://ServerName or IPaddress:Port/Context/publisher>

It is important not to forget the « publisher » string which designates the publication module.

The user name and password corresponding to the Push'n'See server will have been given to you by the system administrator or, if evaluating the application, by e-mail from KOREM's support team.

When all fields are filled up appropriately, make sure to click on the "Save" button for the parameters to be preserved. You will NOT need to reconfigure the server at each use of this application.

From the « Publish to Push'n'See » interface, the user will be able at any time to:

- ✓ Configure a new Push'n'See server (« Add » button)
- ✓ Modify an existing configuration (« Modify » button)
- ✓ Delete an existing configuration (« Delete » button)

## 2. Publication of a MapWindow

Two choices are offered:

- ✓ Publish a new map: creation of a new Push'n'See project based on the current map window.
- ✓ Update an existing project: selection of an existing project to be updated based on the current map window.

Select the appropriate option.

## 3. Data license agreement

By checking this option the user accept the conditions as described in the dialog box. Publication can only proceed if the option is checked.

## 4. Analyse of the workspace

Check this option for the workspace to be analysed prior to publication. This allow the user to confirm that its properties and content are compatible with the application. The analysis will be launched as soon as the "Publish" button is clicked. The user will be asked to save the analysis report to the desired directory, then the analysis itself will be carried out. Once completed, the user will be able to consult the result by either reading the content of the MapInfo message window or consulting the saved report. After the analysis the user will have the possibility to carry on with the publication or to cancel the process to bring modifications to the workspace.

## 5. Publish

If you have chosen to continue with the publication process, you will be redirected to the corresponding window of either of the regular publication or update process. Proceed as for a standard publication or update.

